

#23  
S. Sand  
3/5/00  
PATENT

ATTY. DOCKET NO. IBM/162  
Confirmation No. 9157

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant : Gregory P. Andrews et al. Art Unit: 2152  
Serial No. : 08/818,158 Examiner: Thong Vu  
Filed : March 14, 1997  
For : A COMPONENT DOWNLOAD SELECTION MECHANISM FOR  
WEB BROWSERS

Cincinnati, Ohio 45202

February 19, 2002

Assistant Commissioner for Patents

**ATTENTION: Board of Patent Appeals and Interferences**  
Washington, D.C. 20231

**TRANSMITTAL OF APPEAL BRIEF (PATENT APPLICATION-37CFR 191)**

- Transmitted herewith in triplicate is the APPEAL BRIEF in this application with respect to the Notice of Appeal received by the Office on January 18, 2002.
- STATUS OF APPLICANT**

This application is on behalf of

XX other than a small entity

\_\_\_\_\_ small entity

Verified Statement:

\_\_\_\_\_ attached

\_\_\_\_\_ already filed

03/04/2002 L JOHNSON 00000005 08818158

01 FC:120

3. **FEE FOR FILING APPEAL BRIEF**

Pursuant to 37 CFR 1.17(f) the fee for filing the Appeal Brief is:

\_\_\_\_\_ Small entity \$160.00

XX Other than a small entity \$320.00

RECEIVED  
2002 FEB 27 PM 2:30  
BOARD OF PATENT APPEALS  
AND INTERFERENCES

4. **EXTENSION OF TIME**

Applicant petitions for an extension of time under 37 C.F.R. 1.136(a) for the total number of months checked below:

<u>Months</u>	<u>Fee for other than small entity</u>	<u>Fee for small entity</u>
_____ one month	\$ ..... 110.00	\$ ..... 55.00
_____ two months	..... 400.00	..... 200.00
_____ three months	..... 920.00	..... 460.00
_____ four months	..... 1,440.00	..... 720.00
_____ five months	..... 1,960.00	..... 980.00

Fee: \$ \_\_\_\_\_

If an additional extension of time is required, please consider this a petition therefor.

5. **TOTAL FEE DUE**

The total fee due is:

Appeal Brief Fee \$320.00

Extension Fee \_\_\_\_\_

6. **FEE PAYMENT**


XX Attached is a check in the sum of \$320.00.

\_\_\_\_\_ Charge fee to Deposit Account No. 23-3000.

7. **FEE DEFICIENCY**

XX Charge any additional extension fee required or credit any overpayment to Deposit Account No. 23-3000. A duplicate of this transmittal is enclosed.

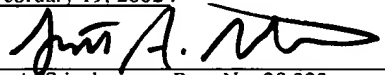
WOOD, HERRON & EVANS, L.L.P.

By   
Scott A. Stinebruner  
Reg. No. 38,323

2700 Carew Tower  
411 Vine Street  
Cincinnati, Ohio 45202  
(513) 241-2324

**CERTIFICATE OF MAILING 37 CFR 1.8**

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to Assistant Commissioner for Patents, Attention: Board of Patent Appeals and Interferences, Washington, D.C. 20231 on February 19, 2002.

  
Scott A. Stinebruner Reg. No. 38,323

Attorney Docket No. IBM/162  
Confirmation No. 9157

PATENT

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

*Ex parte* Gregory P. Andrews  
and Kevin P. Gibson

---

Appeal No. \_\_\_\_\_

Application No. 08/818,158

---

APPEAL BRIEF

---

RECEIVED  
MAR 01 2002  
Technology Center 2600

RECEIVED  
MAR 27 PM 2:20  
UNITED STATES  
PATENT AND TRADEMARK  
OFFICE

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Applicant: Gregory P. Andrews et al.                      Art Unit: 2756  
Serial No.: 08/818,158    Examiner: Thong Vu  
Filed: March 14, 1997    Atty. Docket No.: IBM/162  
For: A COMPONENT DOWNLOAD SELECTION MECHANISM FOR WEB  
BROWSERS

---

**APPEAL BRIEF**

Assistant Commissioner for Patents  
**ATTENTION: Board of Patent Appeals and Interferences**  
Washington, D.C. 20231

**I. REAL PARTY IN INTEREST**

This application is assigned to International Business Machines Corporation, of Armonk, New York.

**II. RELATED APPEALS AND INTERFERENCES**

There are no related appeals or interferences. It should be noted, however, that this is the second appeal instituted in this application. The first appeal, initiated May 2, 2000, resulted in the issuance of a new Office Action by the Examiner in lieu of an Examiner's Answer.

**III. STATUS OF CLAIMS**

Claims 38-75 are pending in the Application. All pending claims stand rejected, and are now on appeal. Claims 1-37 were canceled and claims 38-74 were added by way of amendment in the Amendment and Response filed on November 24, 1999. Claim 75 was added by the Amendment and Response filed on July 9, 2001.

**IV. STATUS OF AMENDMENTS**

There have been no amendments filed subsequent to final rejection.

## V. SUMMARY OF INVENTION

Applicants' invention is generally directed to a dynamic download selection mechanism that permits a user to dynamically and selectively control what contents of a particular file, such as an Internet web page, are downloaded over a computer network.

As discussed, for example, at page 2, line 12 to page 3, line 2, files such as Internet web pages often include references to other components, such as images and executable programs such as Java applets, that are typically downloaded and displayed along with a web page to enhance the visual presentation of information to a user. In fact, it is in part the ability to embed information such as graphical images, photographs, animation, video, sound, etc. into essentially text-based web pages that has made the Internet such a popular entertainment and information medium.

One problem experienced by many users of computer networks such as the Internet, however, is the limited bandwidth available to such users. Particularly when users connect to a network over a dial-up modem, the rate at which information can be transferred to users is severely limited (page 3, lines 17-22).

Given bandwidth limitations, the more information that is associated with a particular web page, the longer that web page takes to download. The inclusion of additional components in a web page only exacerbates the problem, as often the additional components referenced by a web page are larger than the web page itself. And while the provision of additional components along with a web page may significantly enhance a user's viewing experience, in some circumstances, some or all of the additional components referenced by a web page may not be interesting or desirable for a user.

For example, as discussed at page 3, lines 9-15, advertisements are often provided as graphical images and are referenced by particular web pages. Executable components such as Java applets may perform undesirable or unnecessary functions on a user's computer. Other components may not be supported by a user's computer.

In all of the aforementioned situations, the download of such undesirable components is unnecessary. Nonetheless, conventional web browsers are configured to automatically download all components referenced by a web page whenever that web page is downloaded. Thus, while such components are not desired, the user is still often required to wait for those components to

be downloaded. Waiting for unnecessary components to download impedes the browsing process, and thus detracts from a user's browsing experience.

Applicants' invention addresses problems such as these by dynamically prompting a user to select components to be downloaded when a file referencing such components is being downloaded. As shown in Fig. 6, and described in the accompanying text at page 11, line 19 to page 12, line 25, one embodiment of the invention operates by downloading a requested document in response to a user request for the document. When the document is downloaded, the document is parsed to identify references to additional components, and a user is then prompted to select which, if any, of the components should be downloaded. Then, based upon the user's input, selected components are requested and subsequently downloaded.

A user may be dynamically prompted to select components in a number of manners consistent with the invention. As discussed, for example, at page 13, lines 1-15, and as shown in Fig. 7, a list of components may be generated and displayed in a separate "pane" from a primary pane in a window within which the downloaded document is displayed. Each entry in the list may also provide different information about a component, e.g., filename, title, length or size, file type, etc. Moreover, as also shown in Fig. 7, each entry may specify the amount of a component that has already been downloaded, e.g., in terms of raw numbers of bytes or percentage of overall size. Furthermore, Figs. 8 and 9 illustrate other alternative manners of displaying a list, e.g., in a dialog box or embedded within the primary web page, among other alternatives.

As a result, embodiments of the invention permit a user to select what components are or are not downloaded in response to download of a particular file that references such components. Particularly in bandwidth-sensitive environments, such functionality gives users significantly greater control over the speed of Internet accesses, as well as the types of content displayed on the users' computers.

## VI. ISSUE

Whether claims 38-75 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,006,034 to *Heath et al.* in view of U.S. Patent No. 6,004,382 to *Martino*.

## VII. GROUPING OF CLAIMS

For the purposes of appeal, the following groupings of claims are considered to be separately patentable, with the individual claims within each claim grouping standing or falling together:

Group A: claims 38-39, 41, 44-47, 50, 52, 55-58, 61-65, 67 and 69-71

Group B: claims 40, 51, 68 and 73

Group C: claim 74

Group D: claim 75

Group E: claims 42-43, 53-54 and 66

Group F: claims 48-49, 59-60 and 72

## VIII. ARGUMENT

Applicants respectfully submit that the Examiner's rejections of claims 38-75 are not supported on the record, and the rejections should be reversed.<sup>1</sup> The reasons for reversing the Examiner's rejections are presented in greater detail below.

**A. The Group A claims (claims 38-39, 41, 44-47, 50, 52, 55-58, 61-65, 67 and 69-71) were improperly rejected as being unpatentable over *Heath et al.* in view of *Martino*.**

Claims 38-39, 41, 44-47, 50, 52, 55-58, 61-65, 67 and 69-71 were rejected on the basis of 35 U.S.C. §103(a) as being obvious in view of *Heath et al.* in view of *Martino*. A *prima facie* showing of obviousness, however, requires that the Examiner establish that the differences between a claimed invention and the prior art "are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art." 35 U.S.C. §103(a). Such a showing requires that all claimed features be disclosed or suggested by the prior art. Such a showing also requires objective evidence of the suggestion, teaching or motivation to combine or modify prior art references, as "[c]ombining prior art

---

<sup>1</sup>It should be noted for the record that the present rejection of the claims is the Examiner's third such rejection, Applicants having overcome a first rejection based upon a reference to *Thompson et al.* in an Appeal Brief filed June 30, 2000, and having overcome a second rejection based upon a reference to *Gosling* in a Response filed January 29, 2001.

references without evidence of such a suggestion, teaching or motivation simply takes the inventor's disclosure as a blueprint for piecing together the prior art to defeat patentability -- the essence of hindsight." In re Dembiczak, 50 USPQ2d 1614, 1617 (Fed. Cir. 1999).

Applicants respectfully submit that *Heath et al.* and *Martino* do not disclose or suggest, alone or in combination, the various features recited in the Group A claims, and as such, the rejections thereof should be reversed.

Claim 38, which is representative of the Group A claims, recites an apparatus comprising at least one processor, a memory coupled to the at least one processor, and a computer program residing in the memory. Claim 38 further recites the computer program "commencing to download a file referencing a plurality of components," and "dynamically prompting a user to select which of said plurality of components to download."

*Heath et al.* discloses a method of maintaining application programs on client computers in a client-server environment through the use of dynamic component upgrades. Specifically, components of an application program (e.g., executable codes, library files, parameter files and data files) are maintained on a server, with each component having a version identification. A catalog of those components (and their current version id's) is also maintained on the server (*Heath et al.*, col. 1, lines 56-62). A client is then able to initiate an upgrade of a copy of the application program on the client by requesting the catalog file from the server, and then comparing the version id's of its installed components with the version id's stored in the catalog file. For any components that do not match the version id's stored in the catalog file, the current versions of those components are then downloaded from the client to the server, thus updating the client copy of the application program to the current version (*Heath et al.*, col. 1, line 62-col. 2, line 5).

An important aspect of this process is that the upgrade of the application program occurs with as little user intervention, and with as much authenticity and integrity, as is possible. To this extent, it is the version id information for the components stored at the client, and for the components identified in the catalog file, that control what components are ultimately downloaded in *Heath et al.* to upgrade an application program. A user has no option as to which components will be downloaded, only whether or not the upgrade process of checking components and downloading any new/replacement components will proceed. As such, *Heath et*



*al.* can not be read to disclose or suggest "dynamically prompting a user to select which of [a] plurality of components to download" as is required by claim 38.

In this regard, the Examiner admits that *Heath et al.* fails to disclose a computer program having this capability (Final Office Action, page 1). Instead, the Examiner relies on *Martino*, in particular the disclosure at col. 17, line 65 to col. 18, line 17, to allegedly disclose a server that automatically prompts a user to download menu and form data to a transaction entry device.

*Martino*, however, discloses predominantly a forms-based operating system for a data entry terminal, where data is transferred between the terminal and a server via "transactions." A form is downloaded to the entry terminal, and is displayed to a user. One type of form functions as a menu from which a user can select other menus or forms for display (*Martino*, col. 18, lines 2-22). Whenever a form is displayed, a user may enter data, select another form (e.g., another menu) or initiate a process (*Martino*, col. 19, line 44-col. 20, line 56).

However, it is not even entirely clear that a user is dynamically prompted in *Martino* to select which forms should be "downloaded." The cited passage at cols. 17-18 of *Martino* refers to an initial download of a set of forms into a local memory 96 (Fig. 6) that configure a data transaction assembly server (TAS) for use in a particular application. Once a set of forms is downloaded, it appears the forms are retrieved from the local memory in the terminal in response to user input, and not downloaded, as apparently alleged by the Examiner. In some instances, it does appear that additional menus and forms may be downloaded after a user inputs data into a form and sends the form back to a central server (*Martino*, col. 5, lines 15-38). Nonetheless, it is not clear from the reference that this later download is specifically in response to a dynamic prompting of a user when the download of a file has been commenced. Put another way, it is not clear from the reference that a user is dynamically prompted to select forms to "download", rather than to select forms to be retrieved from a local memory.

In addition, even if *Martino* did disclose the download of forms based upon user selection, Applicants submit that the forms are not analogous to "components" within the context of the invention. Instead, it would appear that the *Martino* forms (and in particular, the menu forms that are apparently the most relevant in the Examiner's mind) are more analogous to simple hypertext documents, rather than components referenced by those documents.

Put another way, Applicants define components in the specification as "any item referenced in [an] HTML page to be downloaded and integrated with the page, such as graphics images, background images, audio, video and multimedia files, forms, applets, etc." (Application, page 7, lines 12-15, *emphasis added*). It does not appear, however, that any forms referenced by a menu form in *Martino* are ever "integrated" with such a form. Instead, the more likely reading is that a menu form in *Martino* is used to initiate the display of another form in much the same as a link in a hypertext document is used to display another document. In the former instance, the user selects a menu selection to view a referenced form, and in the latter instance, the user selects a link to view a linked-to document.

Applicants therefore respectfully submit that *Martino* does not disclose or suggest the dynamic prompting of "components" referenced by a file within the context of the invention. Consequently, Applicants also submit that like *Heath et al.*, *Martino* does not disclose or suggest the claimed feature of "dynamically prompting a user to select which of [a] plurality of components to download." And as the combined teachings of the references fails to disclose this claimed concept, Applicants respectfully submit that claim 38 is novel and non-obvious over the prior art of record.

Furthermore, even if a user were dynamically prompted in *Martino* to select forms to be downloaded, and even if the forms disclosed in *Martino* were analogous to "components" within the context of the invention, Applicants respectfully submit that the Examiner's rejection still fails to establish a *prima facie* case of obviousness, as there has been no objective evidence presented of any motivation to combine the references in the manner suggested by the Examiner. Absent any such objective evidence, the Examiner's rejection amounts to nothing more than improper hindsight-based analysis.

As admitted to by the Examiner, *Heath et al.* does not disclose dynamically prompting a user as to which components of an application should be downloaded. In contrast, a downloadable catalog file is used by a client computer in *Heath et al.* to determine which components need to be downloaded when updating an application. Such a configuration is entirely logical in a system such as *Heath et al.*, since application components are typically designed specifically to work together in a particular manner. Were a user given the opportunity to select which components should be downloaded, it is entirely possible that inconsistent

components could be utilized together, causing an application to crash and/or develop incorrect and/or indeterminate results.

Thus, it should be evident from *Heath et al.* that user participation in a determination of what application components should be downloaded during an update would be entirely inconsistent with the goals and objectives of the automated application update process disclosed in *Heath et al.*

In addition, contrary to the Examiner's position, *Martino* would not motivate one of ordinary skill in the art to incorporate user participation into the *Heath et al.* application update process. *Martino*'s forms-based transaction system has no relevance to *Heath et al.*'s application update process. In fact, one important purpose for implementing a forms-based system in *Martino* is to avoid the problems associated with having to update client-based applications. *Martino* avoids these problems through the use of a generalized server-based system where complete sets of forms, rather than applications, are downloaded to general purpose terminals whenever users wish to configure terminals for specific purposes (*Martino*, col. 4, line 66- col. 5, line 14). Moreover, there is no disclosure in *Martino* that any downloaded forms are retained and later selectively updated. Instead, it appears that a new, complete set of forms is downloaded whenever a terminal is to be configured for use in a specific application.

Given the significantly different goals of *Heath et al.* and *Martino*, Applicants respectfully submit that one of ordinary skill in the art would not look to *Martino* for motivation to modify *Heath et al.* to incorporate dynamic prompting of a user.

It is also well settled that a proposed modification to the teachings of a prior art reference cannot render that reference unsatisfactory for its intended purpose or change the principle of operation of that reference. MPEP §§2143.01 and 2145.X.D. Here, modifying *Heath et al.* to prompt a user as to which components should be downloaded runs expressly counter to the goals and objectives of *Heath et al.*

First, dynamically prompting a user to select application components in *Heath et al.* to download would in the least significantly alter the purpose and use of a catalog file, and at the most, eliminate the need to incorporate versioning information in a catalog file, as is disclosed in *Heath et al.* In either event, the principle of operation of the *Heath et al.* arrangement would be significantly different, as the decision on what components needed to be downloaded in order to

properly update an application would essentially be shifted from the application provider to the end user.

Second, dynamically prompting a user to select application components would introduce the possibility of human error, and the attendant risk that incompatible application components from different versions of an application would be installed in a client computer. Given the strong desire in *Heath et al.* of maintaining the integrity and authenticity of an application (see, e.g., col. 2, lines 37-45), any attempt to relinquish control over what components are downloaded would effectively render the *Heath et al.* system unsatisfactory for its intended purpose of providing reliable updates to an application program.

Third, one goal of *Heath et al.* is minimizing processing overhead and user involvement through automating an application update process to the greatest extent possible. See, e.g., col. 1, lines 50-55 ("By intelligently and automatically selecting to download and update only the needed and changed components of an application program, the present method alleviates the concerns of time and efficiency in any client-server network environment which requires highly dynamic application updates."). Adding user prompting to such a scenario would not be consistent with these goals, and as such Applicants respectfully submit that there would be no motivation for one of ordinary skill in the art to modify *Heath et al.* to incorporate dynamic prompting of a user.

Based upon the foregoing, Applicants respectfully submit that the only motivation relied upon by the Examiner in combining *Heath et al.* and *Martino* comes from Applicants' disclosure. As such, the Examiner's rejection improperly relies on hindsight, and a *prima facie* case of obviousness has thus not been established as to claim 38.

As a final matter, Applicants wish to briefly address an apparent misconception on the part of the Examiner in the Examiner's Response to Arguments in ¶19 of the Final Office Action (pages 3-5). In particular, the Examiner argues that:

"Heath taught the client downloads a list of components from server. However Heath fails to detail how to display the list of components." (Final Office Action, page 4, lines 21-22).

It appears from this passage that the Examiner is of the belief that *Heath et al.* discloses dynamic prompting of a user, but that the reference does not how that user is presented with the list of components that can be downloaded. Based upon the discussion of *Heath et al.* presented above, it should be clear that this interpretation is not correct. The "client" that downloads components in *Heath et al.* is an application, and not a user. Moreover, there is no disclosure whatsoever that the download of a catalog file by the *Heath et al.* client, and the subsequent comparison of version information and updating of application components, involves any user interaction. It appears that, at the most, a user may be given the opportunity to manually initiate an update process by the client; however, once that process is initiated, the user provides no further input, and certainly provides no selection of which components should be downloaded. Furthermore, since the goal of *Heath et al.* is to update a program "invisibly in the background" (*Heath et al.*, col. 3, line 6), there would be no reason to even display the contents of the catalog file to the user prior to updating any components.

Applicants therefore respectfully submit that claim 38 is non-obvious over the prior art of record. Reversal of the Examiner's rejections of this claim and of the other Group A claims (claims 38-39, 41, 44-47, 50, 52, 55-58, 61-65, 67 and 69-71), as well as allowance of all such claims, are therefore respectfully requested.

**B. The Group B claims (claims 40, 51, 68 and 73) were improperly rejected as being unpatentable over *Heath et al.* in view of *Martino*.**

The Group B claims introduce the additional concept that the file the references the plurality of components is a hypertext markup language (HTML) document. However, neither *Heath et al.*, nor *Martino*, disclose or suggest that the entities arguably analogous to "files" in the respective references can be HTML documents.

The Examiner argues that *Heath et al.* discloses the use of an HTML document, citing col. 7, lines 35-55 (Final Office Action, ¶6), as well as Fig. 7C and col. 7, line 56 to col. 8, line 14 (Final Office Action, ¶19). However, it is important to note that the cited passages disclose only that an HTML document can be used to provide a link to a server from which a catalog file can be downloaded. There is no disclosure or suggestion in *Heath et al.* that the catalog file itself

(which the Examiner analogizes to a "file referencing a plurality of components") can be an HTML document.

*Martino* does not discuss HTML or hypertext links whatsoever, and the only discussion pertinent to the Internet or other related technologies is based upon the fact that a terminal can be connected to a server over the Internet, and that a terminal can be configured by the forms-based system of *Martino* to operate as an Internet terminal (e.g., for browsing the Internet or reading email) (*Martino*, col. 6, lines 1-14).

As discussed above in the Summary of the Invention, one specific application of Applicants' invention is directed toward addressing the problem of the long download times associated with downloading HTML documents over the Internet, particularly over slow Internet connections such as dial-up modems. By permitting a user to selectively choose, once the download of an HTML document has been commenced, which referenced components in that document should also be downloaded, the time associated with downloading otherwise undesirable components can be avoided, thus reducing the overall time a user has to wait to view of document being downloaded.

As such, the fact that the Group B claims recite the use of an HTML document as the "file referencing a plurality of components" is a patentable distinction over the prior art of record. Neither *Heath et al.*, nor *Martino*, even appreciate the specific problems addressed by Applicants' claimed invention, and as such, these references cannot be read to suggest the dynamic prompting to a user of the components referenced by an HTML document being downloaded. Accordingly, reversal of the Examiner's rejections of the Group B claims (claims 40, 51, 68 and 73), as well as allowance of all such claims, are respectfully requested.

C. **The Group C claim (claim 74) was improperly rejected as being unpatentable over *Heath et al.* in view of *Martino*.**

Claim 74 differs from the Group A claims principally in that the claim recites a method of "downloading an HTML document from a web server to a web browser," where the document includes "references to a plurality of embedded components." Additionally, the claim recites "parsing [an] HTML document for references to said plurality of embedded components" and

"prompting a user to select which of said plurality of embedded components to download by displaying a component download selection list on said web browser."

In rejecting claim 74, the Examiner essentially combines the rejections of claims 38 and 39 (Final Office Action, ¶17), and given that claim 39 recites only that the computer program is a web browser, it is readily evident that the Examiner has failed to properly address the specific limitations of claim 74. The rejection is deficient on its face, and therefore should be reversed.

Moreover, neither *Heath et al.*, nor *Martino*, discloses or suggests either the dynamic prompting of components referenced by an HTML document (see Section VIII.B above), or that such components may be "embedded" in an HTML document. *Heath et al.* discloses a catalog file that provides an inventory of components in an installed application; however, these components are not "embedded" in the catalog file, those components are in fact components of an application program that is separate from the catalog file. Likewise, *Martino* discloses menu forms that arguably reference other forms for display; however, it appears that each form in *Martino* is displayed separately, so any forms that may be analogized to "components" of another form cannot be read to be "embedded" in that other form.

Applicants therefore respectfully submit that the Examiner has failed to establish a *prima facie* case of obviousness as to claim 74. Reversal of the Examiner's rejection, and allowance of this claim, are therefore respectfully requested.

**D. The Group D claim (claim 75) was improperly rejected as being unpatentable over *Heath et al.* in view of *Martino*.**

Claim 75 differs from claim 38 in that the claim additionally recites that the computer program is "further configured to receive user input that selects at least one of the plurality of components, to commence to download at least one selected component from the plurality of components, and to display the file with the selected component embedded therein."

In rejecting the additional features in claim 75, the Examiner relies on *Heath et al.*, col. 7, line 55 to col. 8, line 14 (Final Office Action, ¶18), which generally discusses the initiation of an update process through selection of a hypertext link. This passage, however, is silent as to any manner of displaying a file, much less doing so with any selected components embedded therein.

Moreover, as discussed above in Section VIII.C, neither *Heath et al.*, nor *Martino*, discloses or suggests components that are "embedded" in a file. *Heath et al.* discloses a catalog file that provides an inventory of components in an installed application; however, these components are not "embedded" in the catalog file. Likewise, *Martino* discloses menu forms that arguably reference other forms for display; however, it appears that each form in *Martino* is displayed separately, so any forms that may be analogized to "components" of another form cannot be read to be "embedded" in that other form.

As neither reference discloses or suggests components that are "embedded" in a file, Applicants respectfully submit that neither reference can be interpreted to disclose or suggest displaying a file with a selected component therein, as is required by claim 75. Claim 75 is therefore non-obvious over the prior art of record. Reversal of the Examiner's rejection, and allowance of this claim, are therefore respectfully requested.

**E. The Group E claims (claims 42-43, 53-54 and 66) were improperly rejected as being unpatentable over *Heath et al.* in view of *Martino*.**

The Group E claims additionally recite the concept of displaying a component download selection list in a manner external from the display of the file within which the components identified in the list are referenced. Claim 42, for example, recites that the component download selection list is formed in a second pane of a web browser. Claim 43 similarly recites that the component download selection list is formed in a dialog box.

Neither of these concepts is disclosed or suggested by the prior art of record. With respect to claim 42 (displaying a component download selection list in a second pane of a browser), the Examiner curiously relies on *Heath et al.*, col. 3, lines 20-37. The cited passage refers to the use of a web browser to initiate the download of a catalog file; however, as discussed above, *Heath et al.* cannot even be read to support the proposition that a component download selection list, or even a catalog file, is displayed to a user, so Applicants submit that the reference cannot be relied upon to suggest the display of a list in a pane of a browser. The reference is also utterly silent as to "panes" in a web browser, so Applicants' also fail to appreciate how the reference can be read to disclose or suggest displaying a list in a second pane of a web browser. As such, the Examiner's support for rejecting claim 42 cannot be sustained.



Similarly, with respect to claim 43 (displaying a component download selection list in a dialog box), the Examiner relies on *Martino*, col. 17, line 65-col. 18, line 17, stating that a "dialog box as [sic] inherent feature of menu" (Final Office Action, ¶9). This allegation by the Examiner is completely unsupported by any evidence in the prior art, and Applicants submit that dialog boxes and menus are entirely different user interface components, so it is not in fact inherent that the menu forms of *Martino* would have dialog boxes. To the contrary, *Martino* discloses a preferred 40x25 text display (col. 4, lines 37-45), and there is not even a suggestion in the reference of implementing a graphical, or Windows-type, user interface. As such, the Examiner's support for rejecting claim 43 likewise cannot be sustained.

The rejections of the Group E claims appear to be based principally on hindsight, as the Examiner has failed to establish the obviousness of the additional features recited in these claims based upon the teachings of the prior art. Accordingly, reversal of the Examiner's rejections of the Group E claims (claims 42-43, 53-54 and 66), as well as allowance of all such claims, are respectfully requested.

**F. The Group F claims (claims 48-49, 59-60 and 72) were improperly rejected as being unpatentable over *Heath et al.* in view of *Martino*.**

The Group F claims additionally recite the concept of displaying a status item in a component download selection list that dynamically displays the amount of each of a plurality of components that has been downloaded. As shown, for example, in Figs. 7, 8, and 9, and as described at page 13, line 10, page 14, line 9, and page 14, line 26 of the Application, this claimed feature permits a user to be notified of the amount of each component that has already been downloaded.

In rejecting these claims, other than making mere conclusory statements, the Examiner relies on *Heath et al.*, col. 4, lines 28-48, which is reproduced below for the convenience of the Board:

In an open network architecture, such as the Internet and intranets, the centralized program updating is difficult due to the fact that the individual clients are not necessarily controlled by the server. In the Internet, for example, a Web server communicates with a

remote client on an anonymous basis and cannot easily control the parameters of the client. FIG. 2A illustrates a preferred method of the present invention wherein a client 22 controls the process of a software upgrade in the client utilizing one or more servers 24 on a network. More particularly, in FIG. 2B, a software version upgrade can be initiated through executing an application program on the client 22. The execution command transmits a request signal 23 to the server 24 which holds the latest application components. In the preferred embodiment, the server responds by downloading a catalog of a list of the application components, each identified with the latest version number. Here, the server includes either a single computer or multiple computers or other servers networked together. The catalog file is processed by the client 22 to selectively identify and retrieve required components of the application program from the server 24. (*Heath et al.*, col. 4, lines 28-48).

This passage is silent about displaying status items. Accordingly, the Examiner's rejection is deficient on its face, as the Examiner has provided no support in the prior art for the rejection. Reversal of the Examiner's rejections of the Group F claims (claims 48-49, 59-60 and 72), as well as allowance of all such claims, are therefore respectfully requested.

#### IX. CONCLUSION

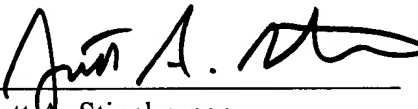
In conclusion, Applicants respectfully request that the Board reverse the Examiner's rejections of claims 38-75, and that the Application be passed to issue. If there are any questions regarding the foregoing, please contact the undersigned at 513/241-2324. Moreover, if any other charges or credits are necessary to complete this communication, please apply them to Deposit Account 23-3000.

Respectfully submitted,

WOOD, HERRON & EVANS, L.L.P.

Date: 19 FEB 2002

2700 Carew Tower  
Cincinnati, Ohio 45202  
(513) 241-2324

By:   
Scott A. Stinebruner  
Reg. No. 38,323

**APPENDIX A: CLAIMS ON APPEAL (S/N 08/818,158)**

1-37. Canceled

1       38.     An apparatus comprising:  
2               at least one processor;  
3               a memory coupled to the at least one processor; and  
4               a computer program residing in the memory, said computer program commencing  
5       to download a file referencing a plurality of components, said computer program  
6       dynamically prompting a user to select which of said plurality of components to  
7       download.

1       39.     The apparatus of claim 38 wherein said computer program comprises a web  
2       browser application.

1       40.     The apparatus of claim 38 wherein said file comprises a hypertext markup  
2       language (HTML) document.

1       41.     The apparatus of claim 38 wherein said computer program includes a component  
2       download selection mechanism, said component download selection mechanism  
3       dynamically creating a component download selection list when said file with said  
4       plurality of components is downloaded.

1       42.     The apparatus of claim 41 wherein said computer program comprises a web  
2       browser and wherein said component download selection list is formed in a second pane  
3       of said web browser and displayed with said file.

1       43.     The apparatus of claim 41 wherein said component download selection list is  
2       formed in a dialog box.

1       44.     The apparatus of claim 41 wherein the component download list is inserted into  
2       said file and displayed to a user with said file.

1       45.     The apparatus of claim 41 wherein said component download selection list  
2       contains the file name for each of said plurality of components.

1       46.     The apparatus of claim 41 wherein said component download selection list  
2       contains the type for each said plurality of components.

1       47.     The apparatus of claim 41 wherein said component download selection list  
2       contains the size of each said plurality of components.

1       48.     The apparatus of claim 41 wherein said component download selection list  
2       includes a status item, said status item dynamically displaying the amount of each of said  
3       plurality of components that has been downloaded.

1       49.     The apparatus of claim 48 wherein said status item includes the percentage of a  
2       component downloaded.

1       50.     A method for downloading a document, the document including a document with  
2       references to a plurality of components, the method comprising the steps of:  
3             a) downloading said document;  
4             b) prompting a user to select which of said plurality of components to download;  
5       and  
6             c) downloading said selected components.

1       51.     The method of claim 50 wherein the document comprises an HTML document.

*Appendix A: Claims on Appeal 08/818,158*

1     52.     The method of claim 50 wherein the step of prompting a user to select which of  
2     said plurality of components to download comprises displaying a component download  
3     selection list.

1     53.     The method of claim 52 wherein said component download selection list  
2     comprises a dialog box.

1     54.     The method of claim 52 wherein said component download selection list  
2     comprises is displayed in a pane in a web browser.

1     55.     The method of claim 52 wherein said component download selection list is  
2     inserted into said document.

1     56.     The method of claim 52 wherein said component download selection list  
2     comprises the file name for each of said plurality of components.

1     57.     The method of claim 52 wherein said component download selection list  
2     comprises the type for each said plurality of components.

1     58.     The method of claim 52 wherein said component download selection list  
2     comprises the size for each said plurality of components.

1     59.     The method of claim 50 wherein said component download selection list  
2     comprises a status item, said status item dynamically displaying the amount of each of  
3     said plurality of components that has been downloaded.

1     60.     The method of claim 59 wherein said status item includes the percentage of a  
2     component downloaded.

1     61.     A program product comprising:

- 2           (A)    a computer program, said computer program commencing to download a  
3 file referencing a plurality of components, said computer program dynamically prompting  
4 a user to select which of said plurality of components to download; and  
5           (B)    signal bearing media bearing said download selection mechanism.

1       62.    The program product of claim 61 wherein said computer program includes a  
2 component download selection mechanism, said component download selection  
3 mechanism dynamically creating a component download selection list when said file with  
4 said plurality of components is downloaded.

1       63.    The program product of claim 61 wherein the signal bearing media comprises  
2 recordable media.

1       64.    The program product of claim 61 wherein the signal bearing media comprises  
2 transmission media.

1       65.    The program product of claim 61 wherein said computer program comprises a  
2 web browser application.

1       66.    The program product of claim 62 wherein said component download selection list  
2 is formed in a dialog box.

1       67.    The program product of claim 62 wherein the component download list is inserted  
2 into said file and displayed to a user with said file.

1       68.    The program product of claim 61 wherein said file comprises a hypertext markup  
2 language (HTML) document.

1       69.    The program product of claim 62 wherein said component download selection list  
2 contains the file name for each of said plurality of components.

1       70.    The program product of claim 62 wherein said component download selection list  
2       contains the type for each said plurality of components.

1       71.    The program product of claim 62 wherein said component download selection list  
2       contains the size of each said plurality of components.

1       72.    The program product of claim 62 wherein said component download selection list  
2       includes a status item, said status item dynamically displaying the amount of each of said  
3       plurality of components that has been downloaded.

1       73.    An apparatus comprising:  
2            at least one processor;  
3            a memory coupled to the at least one processor; and  
4            a web browser application residing in the memory, said web browser application  
5       including a component download selection mechanism, said component download  
6       selection mechanism dynamically creating a component download selection list when an  
7       HTML document with a plurality of components is downloaded, said component  
8       download selection mechanism prompting a user to select which of said plurality of  
9       components to download.

1       74.    A method for downloading an HTML document from a web server to a web  
2       browser, the document including a document with references to a plurality of embedded  
3       components, the method comprising the steps of:  
4            a) requesting said HTML document from said web server;  
5            b) parsing said HTML document for references to said plurality of embedded  
6       components;  
7            c) prompting a user to select which of said plurality of embedded components to  
8       download by displaying a component download selection list on said web browser; and  
9            d) requesting from said web server said selected embedded components.

1    75. (Added) An apparatus comprising:  
2            at least one processor;  
3            a memory coupled to the at least one processor; and  
4            a computer program residing in the memory, said computer program commencing  
5    to download a file referencing a plurality of components, said computer program  
6    dynamically prompting a user to select which of said plurality of components to  
7    download, wherein the computer program is further configured to receive user input that  
8    selects at least one of the plurality of components, to commence to download at least one  
9    selected component from the plurality of components, and to display the file with the  
10   selected component embedded therein.